System-Level Design and Rapid Prototyping for TI DSPs



Ken Karnofsky

The MathWorks, Inc.

DSPSFest 2000

August 4, 2000

MATLAB
SIMULINK

The MATH WORKS Inc.

AGENDA

- DSP Design Process Challenges
- MathWorks System-Level Design Solutions
- Texas Instruments DSP Developer's Kit

Design Challenges

- **Time-to-market pressure**
  - **Shorter product cycles**
  - **Increasing complexity**
  - **Shortage of DSP expertise**
- **Traditional Process Limitations**
  - **Gap between algorithm research and implementation**
  - **Poor integration of component design teams**
  - **System verification/testing occurs too late**

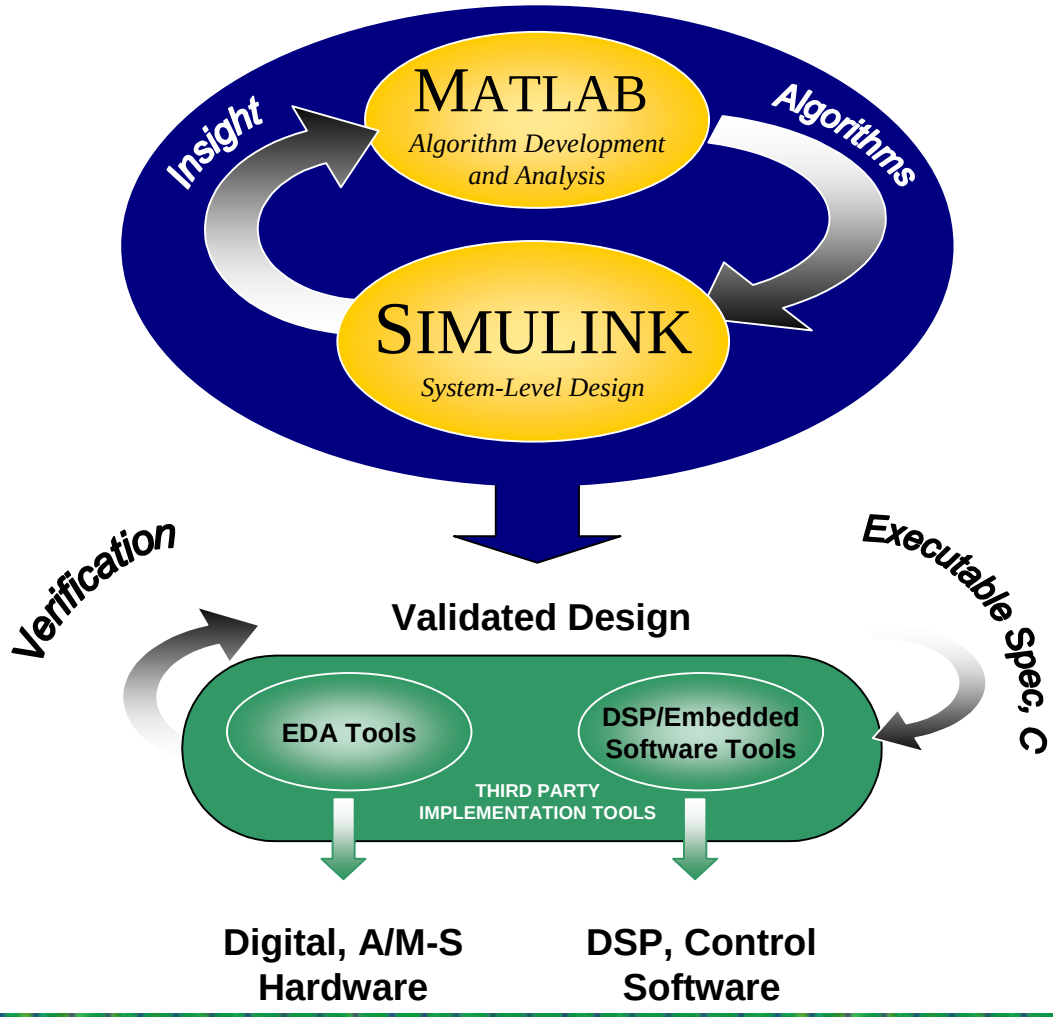
Design Flow Problem

- Ambiguous specifications
- Error-prone manual re-coding
- Design flaws detected too late
- Design failure risk too high

Digital, A/M-S Hardware

DSP, Control Software

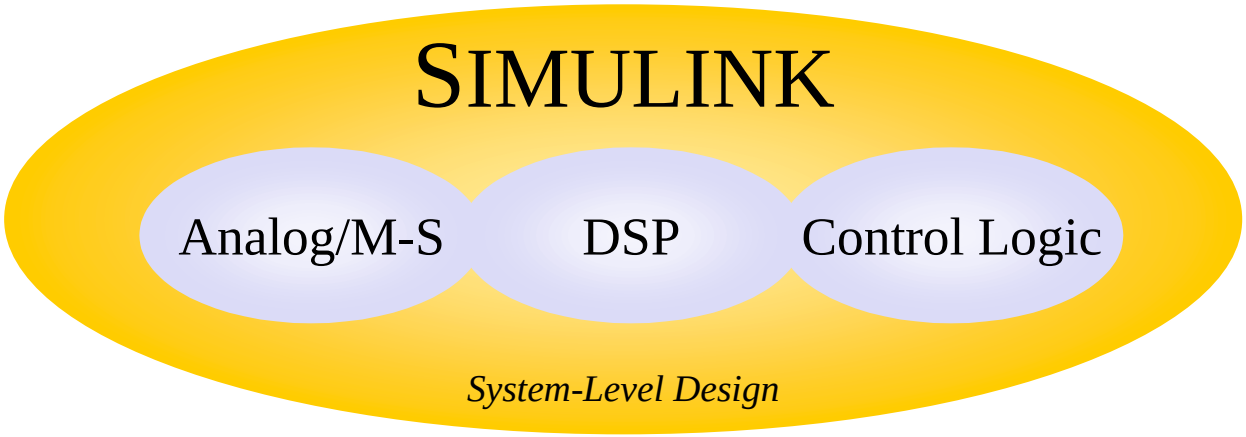
Simulink System-Level Solution



Design Team Integration Problem

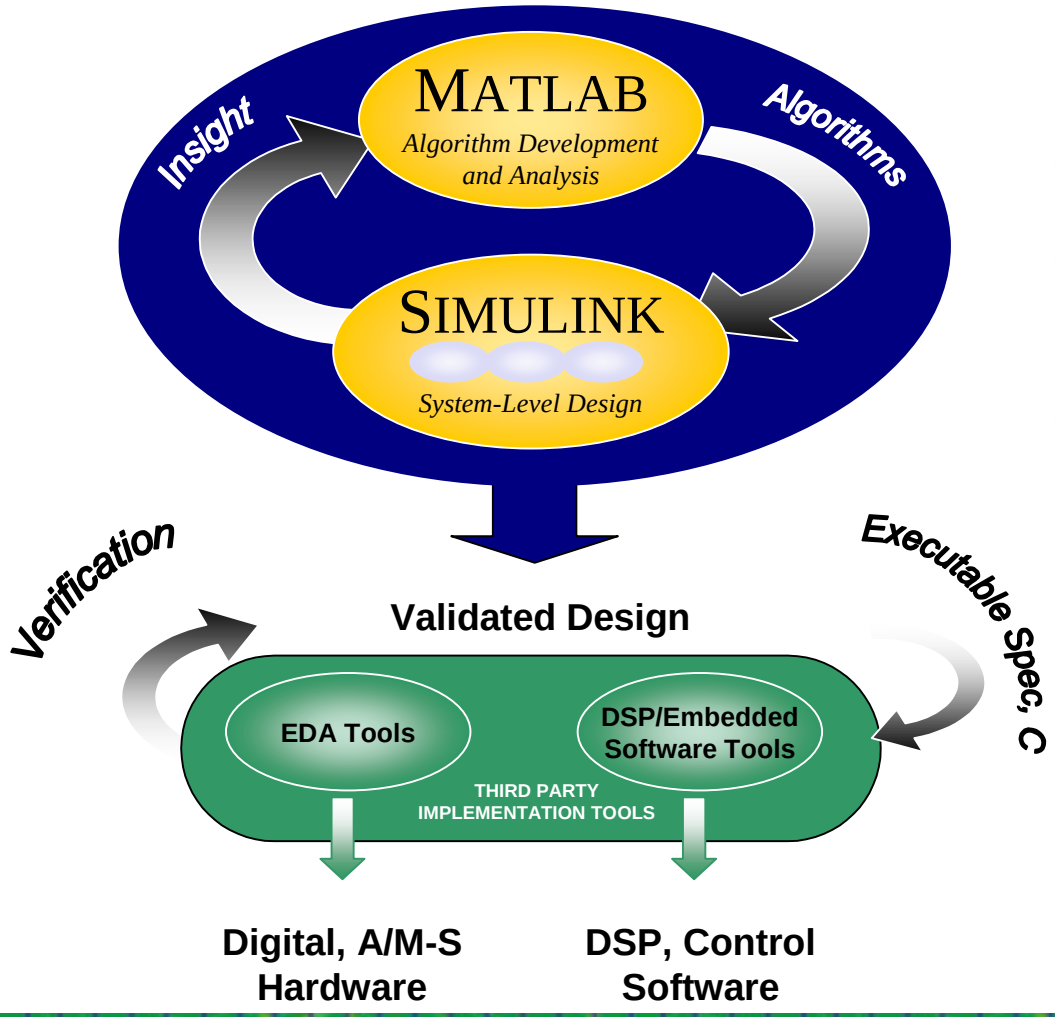
- **Independent design teams and tools**
- **Can’t simulate component interactions**
- **Can’t test whole system**
- **Expensive over-design**

Simulink Integrated Design Solution



- **Common tool for entire design team**
- **Simulate behavior of whole system**
- **Model component interactions**

Simulink Design Environment

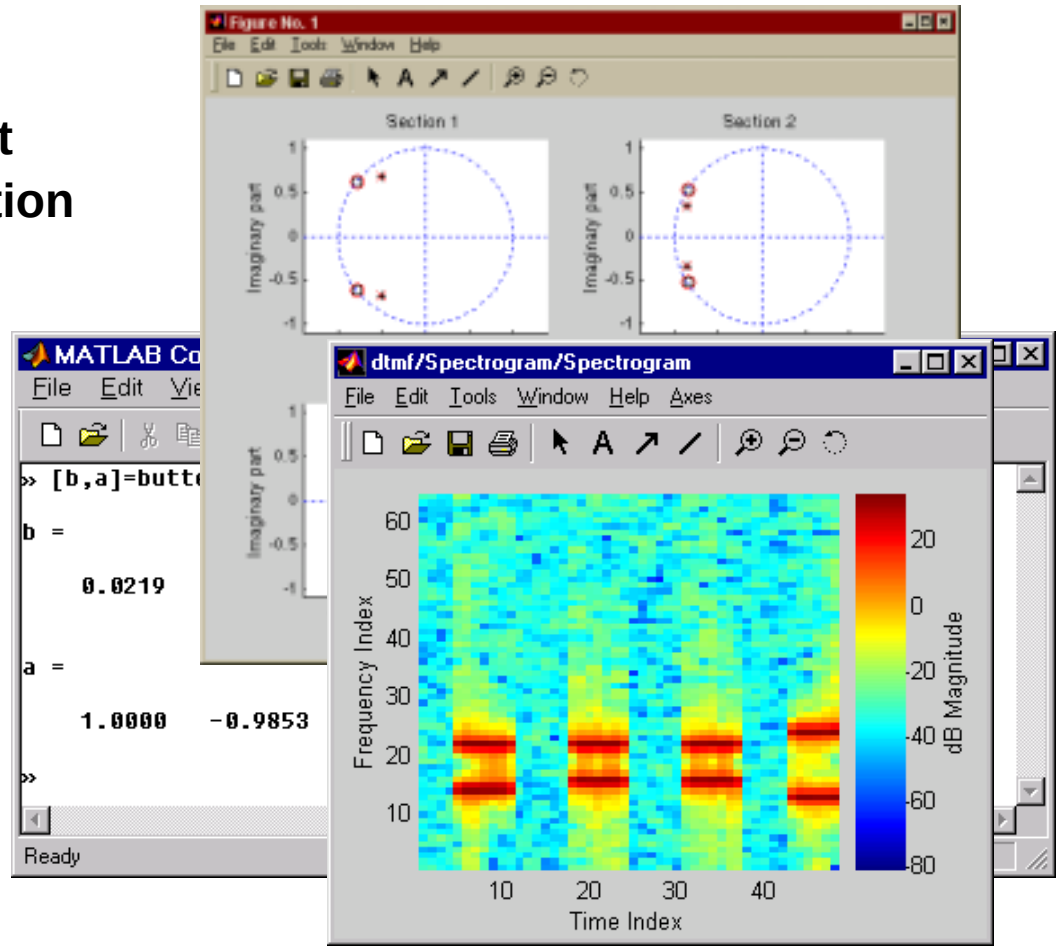


- Test entire system
- Model component interactions
- Reduce design risk
- Reduce time-to-market

The MathWorks System-Level Design Environment

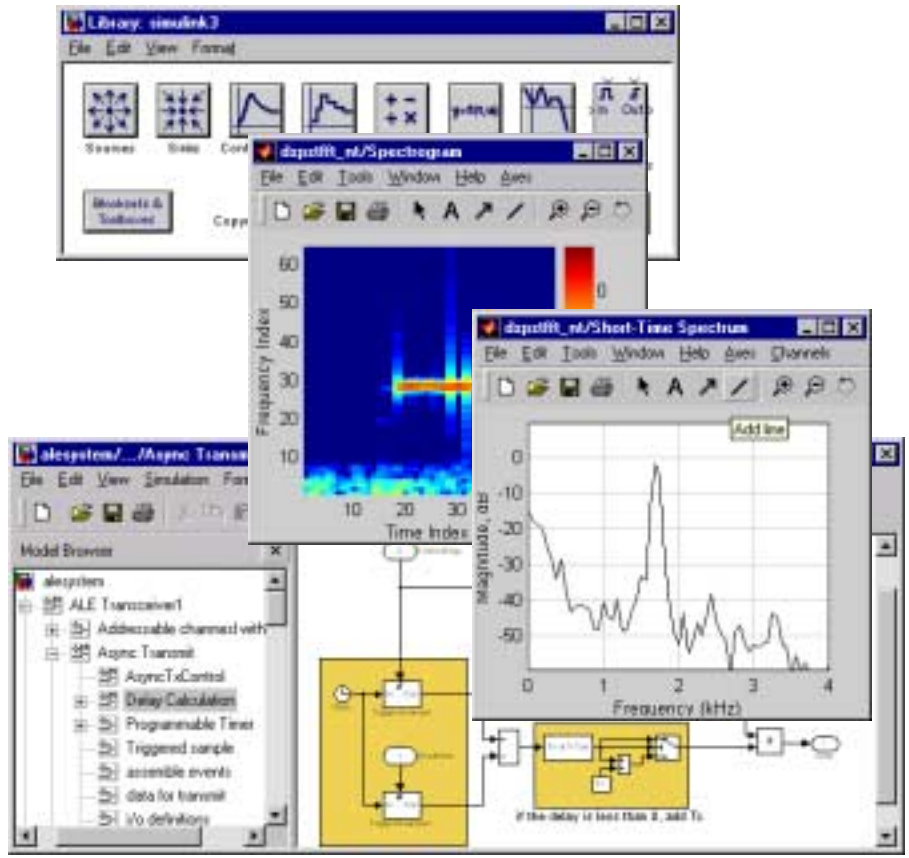
MATLAB - The Language of DSP Algorithms

- MATLAB
  - Algorithm development
  - Analysis and visualization
  - Open and extensible
- MATLAB Toolboxes
  - Signal Processing
  - Quantized Filtering
  - Image Processing
  - Wavelet
  - Control Systems
  - Data Acquisition
  - and many more



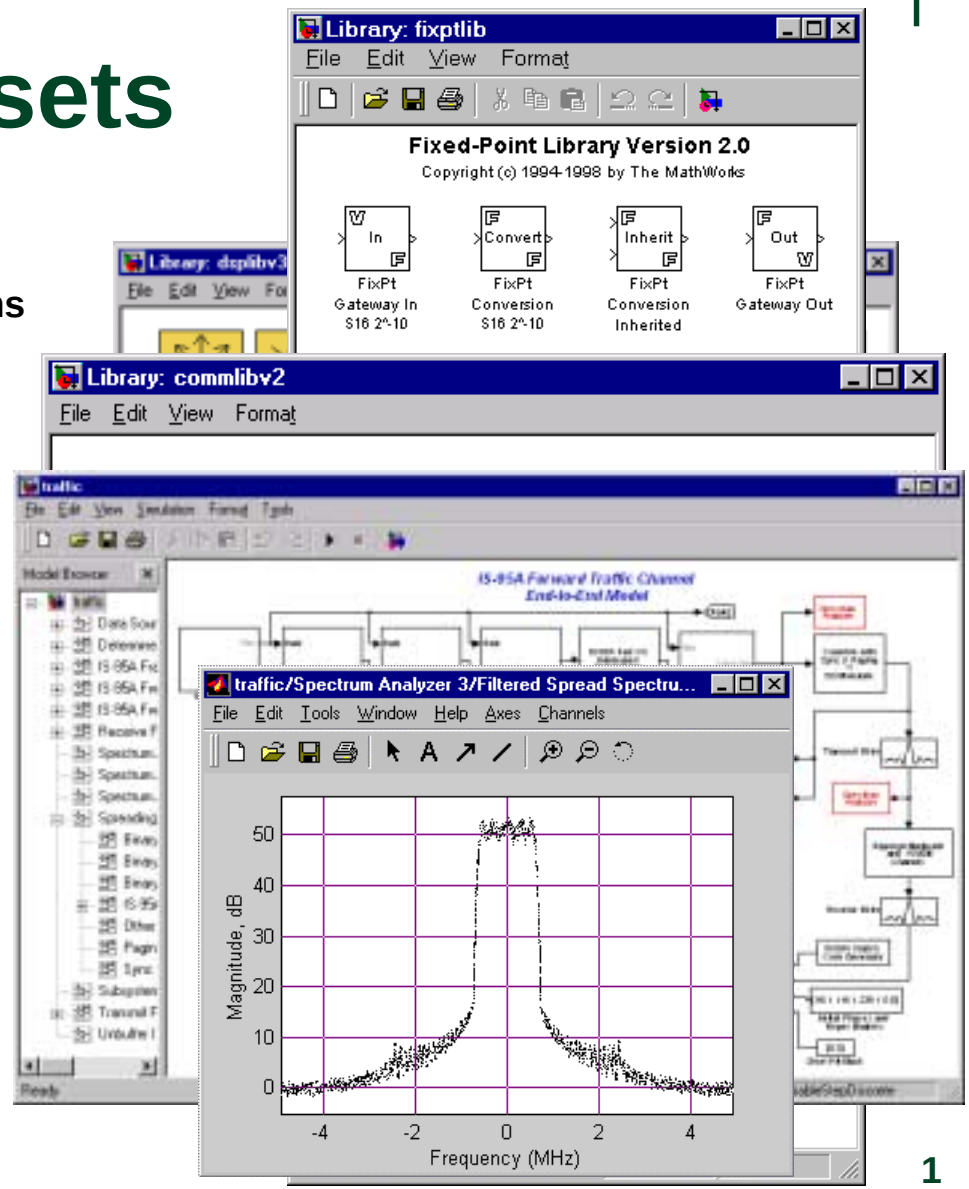
Simulink

- Hierarchical block diagrams
- Dynamic simulation
- Digital, analog/mixed signal
- Visualize signals
- Co-develop with C code
- Rapid prototyping
- Integrated with MATLAB



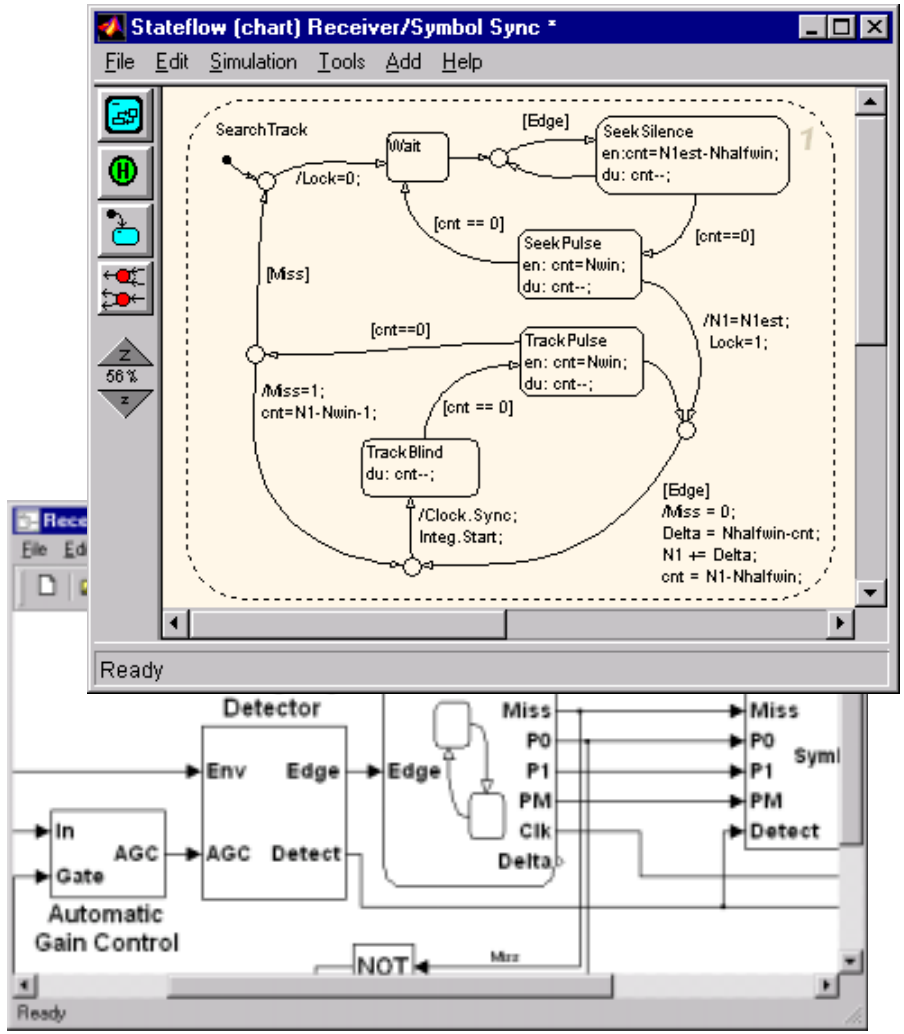
Simulink Blocksets

- **DSP**
  - Filtering, math, estimation, transforms
  - Multi-rate
  - Multi-channel, frame-based
- **Fixed Point**
  - Bit-true 1-128 bits
  - Scaling, rounding and overflow
  - Include own bit-true code
  - Filter wizard
- **Communications**
  - Modulation, coding, synchronization
  - Channel models
  - BER/FER system testing
- **CDMA Reference Blockset**
  - Validated for IS-95A standard
  - Springboard for 3G designs



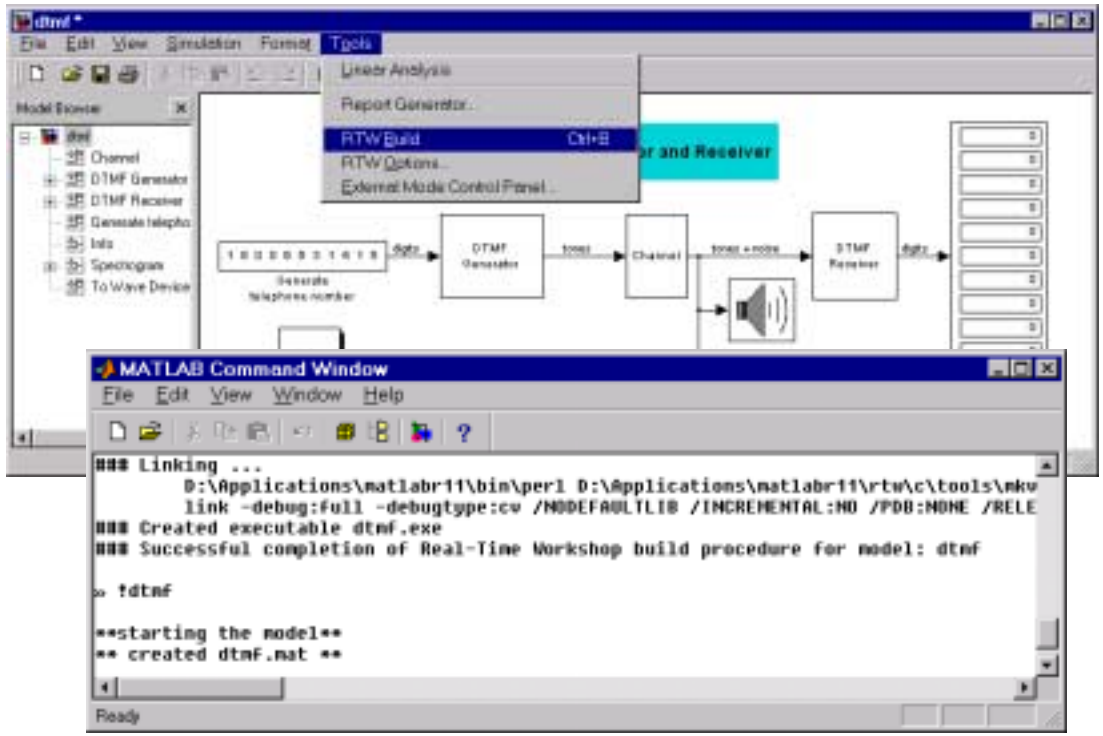
Stateflow

- Integrated with Simulink
- Finite state machines
- Flow diagrams
- Event driven systems
  - Control logic
  - Protocols
  - Synchronization
- Stateflow coder
  - Integer C code generation

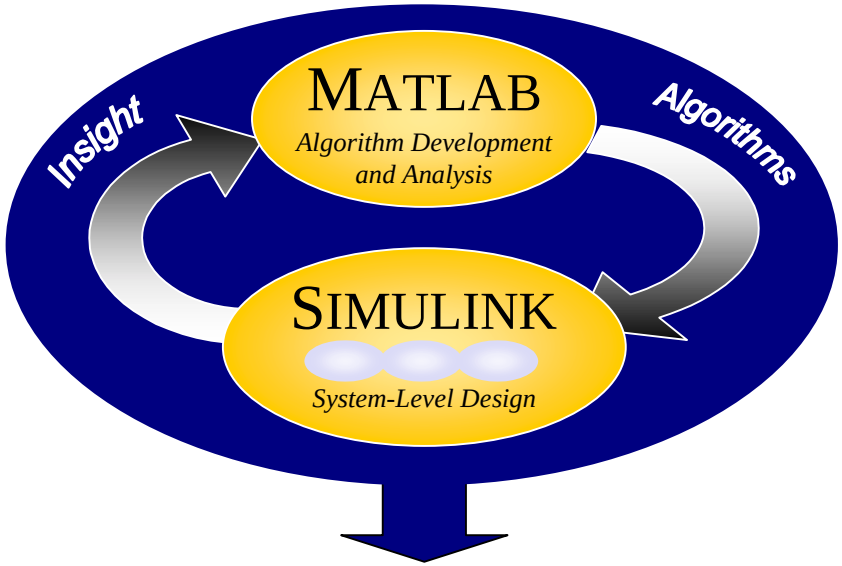


Real-Time Workshop

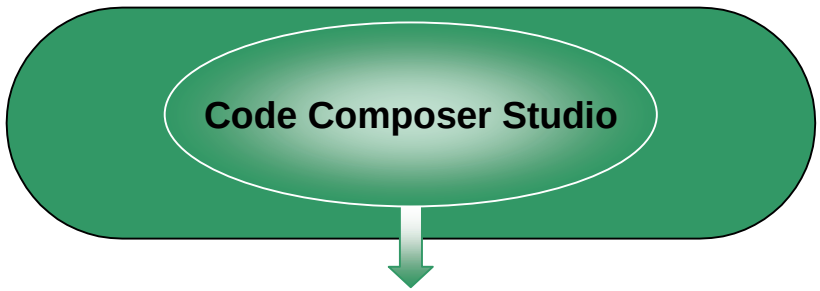
- **Code generation from Simulink**
  - **ANSI C**
  - **Target-specific customization**
- **Real-time rapid prototyping**
- **High performance standalone simulations**



The Next Challenge: Concept-to-Code Integration for eXpressDSP

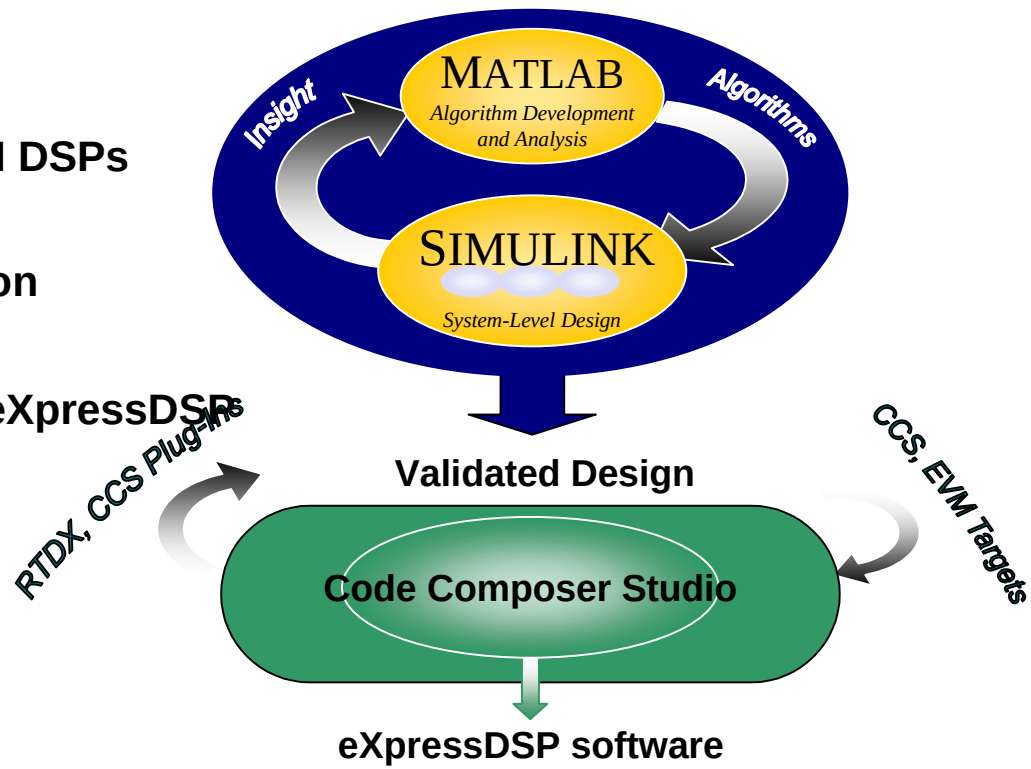


?



TI DSP Developer's Kit

- **Integrates MATLAB and Simulink with eXpressDSP tools**
- **Unified top-down design flow for TI DSPs**
- **Ideal environment for DSP education**
- **MathWorks is 100% committed to eXpressDSP**



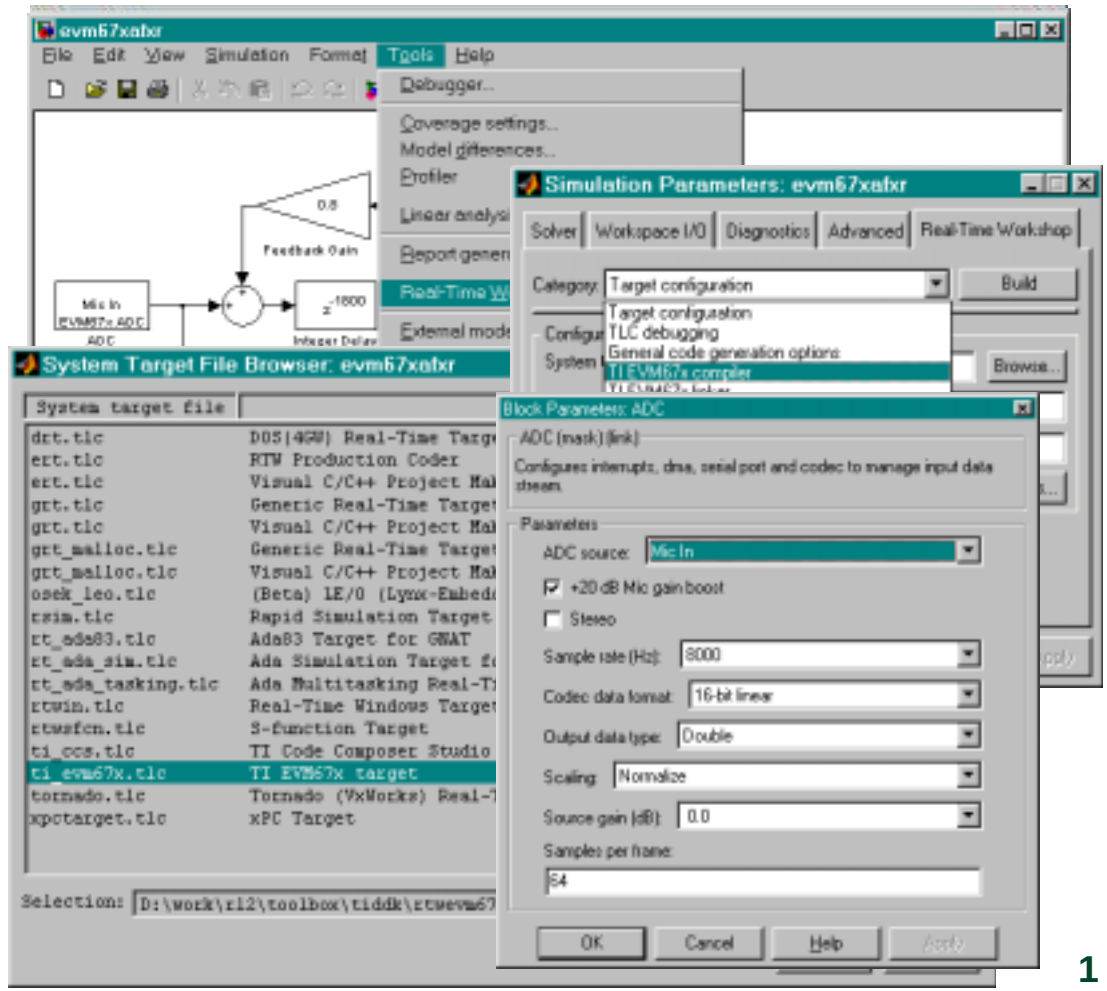
TI DDK Key Features

- **Real-time codegen and rapid prototyping with Simulink**
  - **EVM67x Target**
  - **Code Composer Studio Target**
  - **Benefits**
    - Design graphically in Simulink, deploy and test on TI DSPs
    - Minimize manual algorithm re-coding

- **Real-time analysis and debugging with MATLAB**
  - **Code Composer Studio Automation Interface**
  - **RTDX Interface**
  - **Benefits**
    - Simplify verification and debugging
    - Customize and automate testing, analysis

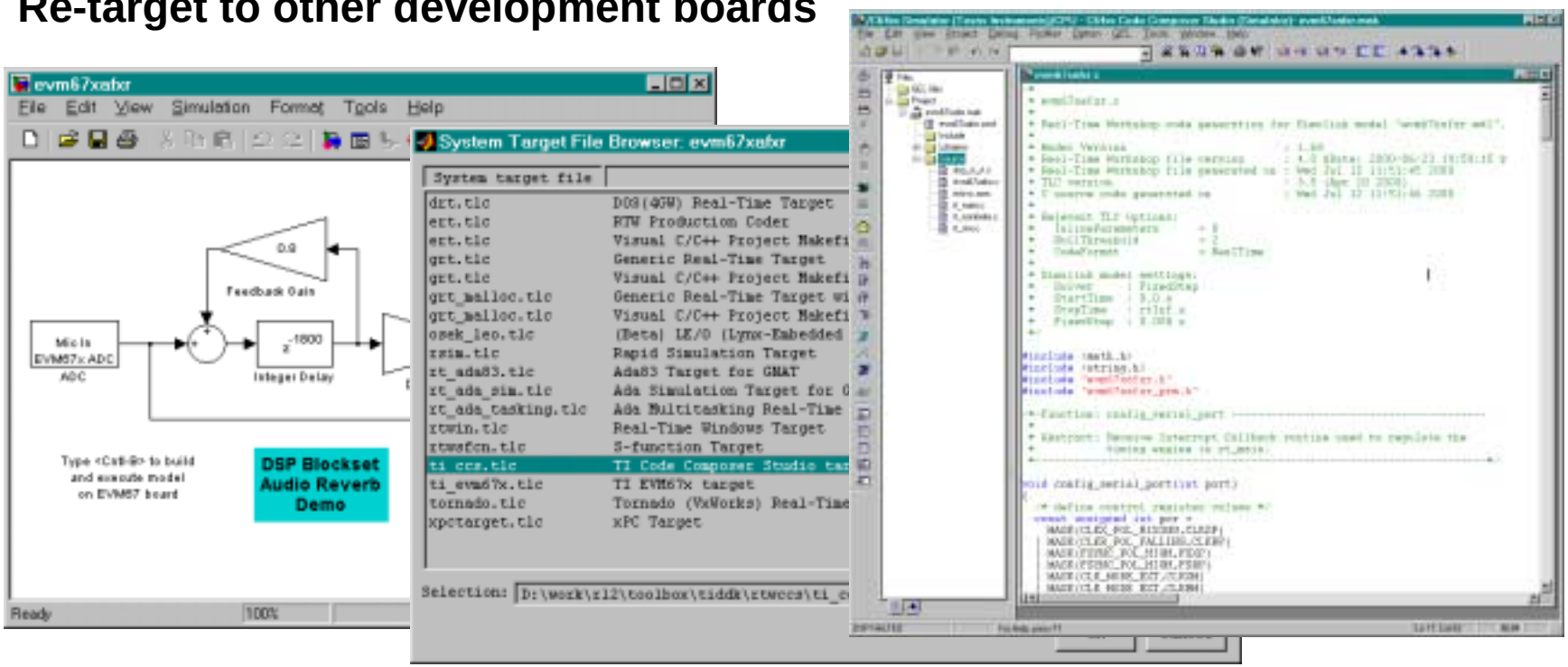
EVM67x Target

- One-button build & run on C6701 EVM
- Real-time algorithm testing
- Full control of board settings
- Includes codec, LED blocks with source code



Code Composer Studio Target

- Automatically generate C code from Simulink
- Automatically build CCS project
- Integrate with custom code
- Re-target to other development boards



MATLAB - eXpressDSP Plug-ins

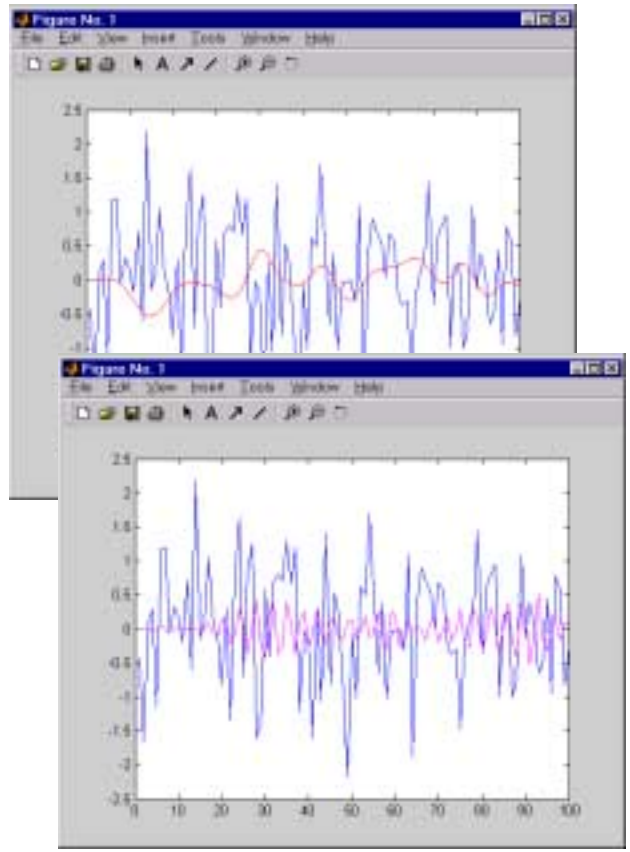
- **Code Composer Studio Plug-in**
  - **Access CCS API from MATLAB**
  - **Compare target outputs to MATLAB or Simulink results**
  - **Integer and floating point data formats**
- **RTDX Plug-in**
  - **Programmatic access to RTDX API from MATLAB**
  - **Real-time analysis and visualization of TI DSP applications**
  - **Create custom tools and GUIs in MATLAB**

Plug-in Example

```
% Activate Link to Code Composer
target=ccsdsp;
target.cccd('D:\Work\RTDX\Target\DemoFIR');
target.ccdir;
% Load Workspace and Target Code
target.open('DemoFIR.wks');
target.open('DemoFIR.out');

[…]
% Use Matlab to design a low-pass filter and
transfer to DSP
new_coeff = fir1(20,0.2);
target.write(addr_coeff{2},new_coeff);

% Run the filter and read the results
target.run('runtohalt',20.0);
filt_dout =
target.read(addr_dout{2},size(test_din),'double');
plot(1:100,test_din,'b-',1:100,filt_dout,'r-');
```



Summary

- Simulink system-level DSP design solutions
- New eXpressDSP plug-in shipping in Q4
- Top-down design flow for TI DSPs
- Integrates tools from industry leaders
- Bridges the gap between algorithm development and implementation